

Gates & Cooper *LLP*

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045

RECEIVED

MAY 21 2003

Technology Center 2100

FAX TRANSMISSION TO USPTO

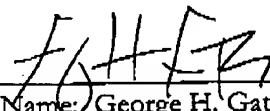
TO: Commissioner for Patents
Attn: Examiner William L. Bashore
Patent Examining Corps
Facsimile Center
Washington, D.C. 20231

FROM: George H. Gates
OUR REF.: ST9-98-052
TELEPHONE: (310) 642-4146

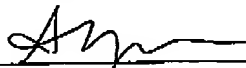
Total pages, including cover letter: **72**PTO FAX NUMBER: **(703) 746-7238**

If you do NOT receive all of the pages, please telephone us at (310) 641-8797, or fax us at (310) 641-8798.

Title of Document Transmitted:	BRIEF OF APPELLANTS INCLUDING APPENDIX, AND COPY OF INTERVIEW SUMMARY
Applicant:	Howard J. Glaser et al.
Serial No.:	09/162,685
Filed:	September 29, 1998
Group Art Unit:	2176
Our Ref. No.:	ST9-98-052

By: 
Name: George H. Gates
Reg. No.: 33,500

I hereby certify that this paper is being transmitted by facsimile to the U.S. Patent and Trademark Office on the date shown below.


Signature

2/28/03
Date

GHG/sjm

G&C 30571.179-US-01

Due Date: March 2, 2003

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of:)	
)	
Inventor: Howard Justin Glaeser, et. al)	Examiner: William L. Bashore
)	
Serial #: 09/162,685)	Group Art Unit: 2176
)	
Filed: September 29, 1998)	Appeal No.: _____
)	
Title: HTML MAPPING SUBSTITUTION)	
GRAPHICAL USER INTERFACE FOR)	
DISPLAY OF ELEMENTS MAPPED)	
<u>TO HTML FILES</u>)	

BRIEF OF APPELLANTS

Commissioner for Patents
Washington, D.C. 20231

Dear Sir:

In accordance with 37 CFR §1.192, Appellants hereby submit the Appellants' Brief on Appeal from the final rejection in the above-identified application, in triplicate, as set forth in the Office Action dated October 2, 2002.

I. REAL PARTY IN INTEREST

The real party in interest is International Business Machines Corporation, the assignee of the present application.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences for the above-referenced patent application.

III. STATUS OF CLAIMS

Claims 1-34 are pending in the application.

Claims 2, 13, and 24 are objected to under 37 CFR 1.75(c) for failing to further limit the subject matter of a previous claim.

Claims 1-8, 11-19, 22-30, and 33-34 are rejected under 35 U.S.C. §103(a) as being unpatentable over Foley et al., U.S. Patent No. 5,706,502 (Foley), in view of Arora et al., U.S. Patent No. 5,911,145 (Arora), and in view of Francis et al., U.S. Patent No. 6,182,092 (Francis).

Claims 9-10, 20-21, 31-32 are rejected under 35 U.S.C. §103(a) as being unpatentable over Foley, Arora, and Francis as applied to claims 1, 12, and 23 above, and further in view of Lisle et al., U.S. Patent No. 6,069,630 (Lisle).

IV. STATUS OF AMENDMENTS

No amendments to the claims have been made subsequent to the final Office Action.

V. SUMMARY OF THE INVENTION

Appellant's invention, as recited in independent claims 1, 12, 23, and 34, is generally directed to a displaying a relationship between an HTML file and an element from a form. Specifically, a mapping/relationship is displayed in a graphical user interface. The mapping is between an HTML file and an element from a form, wherein the element was transferred from the form to an HTML page.

The graphical user interface indicates the relationship between the element, the form that the element was transferred from, and an HTML file that the element was transferred to. Such claim language (in the independent claims) illustrates that the user may easily view the relationships of objects involved in the creation of a web page in a graphical user interface. Such viewing capability may further allow a user to easily manipulate a page and various elements in a page using a tool for building an HTML page (associated with an HTML file). Further, as described in the specification, since the form contains one or more elements, and many instances of an element may be used in various HTML pages, only one copy of the form (that includes the element) needs to be retrieved locally (see page 15, lines 1-5). Accordingly, processing and transfer time are optimized.

Once the information is read from the project file that contains the relationship between the element and a particular HTML file, the information is processed to obtain a mapping from the element in the form to the HTML file. Once obtained, the mapping is displayed in a graphical user interface. The graphical user interface allows the user to identify the element in the HTML file, the form where the element was transferred from, and the HTML file where the element was transferred to.

Dependent claims 2, 13, and 24 provide a timing element that specifies when the information in the independent claims is generated. Specifically, the information is generated when the element is transferred from the HTML page associated with the HTML file.

Dependent claims 3, 14, and 25 provide the element from the independent claims is selected from a group that comprises a visual control and a non-visual. Further, the non-visual control is selected from a group comprising a button, a picklist, and a data entry box.

Dependent claims 4, 15, and 26 specify that the information comprises an element name and an HTML file name.

Dependent claims 5, 16, and 27 depend on claims 4, 15, and 26 and further add the limitation that the information also comprises a form name.

Dependent claims 6, 17, and 28 provide details regarding how to display the mapping. Specifically, these claims provide that the mapping is displayed by presenting an element name and an HTML file name in a row of a table.

Dependent claims 7, 18, and 29 further elaborate on the table of claims 6, 17, and 28. Specifically, these claims indicate that the table has cells that are defined by row and column and the mapping is entered into the cell.

Dependent claims 8, 19, and 30 provide alternative details regarding how to display the mapping. Specifically, these claims provide that the mapping is displayed by presenting an element name and the HTML file name in a column of a table.

Dependent claims 9, 20, and 31 provide the additional capability to flag a mapping (between the element and HTML file) that is invalid.

Dependent claims 10, 21, and 32 further elaborates the flagging of claims 9, 20, and 31. Specifically, a project file is read and the HTML file name is extracted from the project file.

Thereafter, the HTML file name is used to search for the HTML file. If the HTML file is not found, the mapping is flagged as invalid.

Dependent claims 11, 22, and 33 provide for modifying the mapping. Specifically, a modified mapping is accepted and then stored in a project file.

VI. ISSUES PRESENTED FOR REVIEW

Whether claims 2, 13, and 24 are objectionable under 37 CFR 1.75(c) as being of improper dependent form for failing to further limit the subject matter of a previous claim.

Whether claims 1-8, 11-19, 22-30, and 33-34 are unpatentable under 35 U.S.C. §103(a) as being rendered obvious over Foley in view of Arora and in view of Francis.

Whether claims 9-10, 20-21, 31-32 are unpatentable under 35 U.S.C. §103(a) as being rendered obvious over Foley, Arora, Francis, and further in view of Lisle.

VII. GROUPING OF CLAIMS

The rejected claims do not stand or fall together. Each claim is independently patentable. Separate arguments for the patentability of each claim are provided below.

VIII. ARGUMENTS

A. Claims 2, 13, and 24 Further Limit The Subject Matter Of The Independent Claims From Which They Depend.

To fully understand this rejection, a brief history relating to the rejection is useful. On September 22, 2000, a first Office Action was mailed to Applicant (referred to as OA1). In response, Appellants amended the independent claims to incorporate some of the limitations of dependent claims 2, 13, and 24. Specifically, the independent claims were amended to incorporate the limitation that the element was "transferred from a form to an HTML page and an HTML file associated with the HTML page". However, the timing element regarding when the information was generated was not incorporated into the independent claims (e.g., dependent claim 2 provides that "the information is generated when the element is transferred from the form to the HTML page associated with the HTML file").

A Final Office Action (referred to as OA2) was then mailed by the PTO on March 22, 2001. In response to the finality of the Office Action, Appellants filed a petition to remove the finality of the Office Action on May 18, 2001. In the Petition, Appellants stated:

The Applicants' amendment of the claims merely amended the independent claims to incorporate dependent claim limitations. For example, the independent claims 1, 12, 23 were amended to include a limitation such that the element was transferred from a from to an HTML page and an HTML file associated with the HTML page, which limitation was incorporation from dependent claims 2, 13, and 24 respectively of the application as originally filed. The original dependent claims 2, 13, and 24 specifically provided "...the element is transferred from a from to the HTML page associated with the HTML file."

Further, Appellants submitted an after final response on May 22, 2001 to OA2. Thereafter, pursuant to various interviews with the Examiner, a draft copy of an Interview Summary (copy enclosed) was mailed from the PTO on August 22, 2001 indicating that the finality of OA2 would be removed.

On September 17, 2001, the PTO mailed a non-final Office Action (referred to as OA3) objecting claims 2, 13, and 24 under 37 CFR 1.75(c). On December 17, 2001, Appellants filed a response traversing the objections. On April 10, 2002, the PTO mailed another non-final Office Action (referred to as OA4) disagreeing with Appellants traversal. Appellants filed a response on July 10, 2002 restating the traversal of the objection and further requesting the Examiner provide specific reference to the language in the independent claims that provide the timing element as to when the information is generated as recited in dependent claims 2, 13, and 24.

A Final Office Action was mailed by the PTO on October 2, 2002 continuing the objection. The Final Office Action also responds to Appellants prior submissions as follows:

Applicant continues to traverse the Examiner's objection of claims 2, 13, 24 under CFR 1.75(c), as being of improper dependent form for failing to further limit the subject matter of a previous claim. Applicant argues on pages 8-9 (section V – Non Art Rejection) of the amendment that dependent claims 2, 13, 24 provide a timing element, therefore providing a further limitation of the independent claims. The Examiner respectfully disagrees. The Examiner notes that Applicant appears to contradict his own position set forth in Petition Under 37 C.F.R. 1.181 to invoke Supervisory Authority (filed 5/21/2001 as paper #5). Page 2 of said petition states in part:

The Applicants' amendment of the claims merely amended the independent claims to incorporate dependent claim limitations. For example, the independent claims 1, 12, 23 were amended to include a limitation such that the element was transferred from a from to an HTML page and an HTML file associated with the HTML page, which limitation was incorporation from dependent claims 2, 13, and 24 respectively of the application as originally filed.

Using claims 1, and 2 as an example, the relevant limitation of claim 1 states:

"reading information from a project file, the information comprising a relationship between the element that has been transferred from the form to the HTML page and the HTML file associated with the HTML page".

Dependent claim 2 states:

"The method of claim 1, wherein the information is generated when the element is transferred from the form to the HTML page associated with the HTML file."

The Examiner does not see any evidence of claim 2 further limiting the subject matter of independent claim 1. This equally applies to claim 13 (dependent from claim 12), and claim 24 (dependent from claim 23).

Appellants continue to traverse this objection. Specifically, there is no use of the term "generated" in the independent claims. Only the dependent claims provide the limitation regarding when the element is "generated". Thus, the timing element regarding when the information was generated was not incorporated into the independent claims.

Further, Appellants submit that the arguments submitted are not contradictory. The Petition to remove the finality was based on the rationale that some of the dependencies were brought up (and not all of them). In this regard, Appellants continue to submit that the dependent claims provide a limitation regarding when the information is generated. The independent claims do not provide for such a limitation. Accordingly, claims 2, 13, and 24 do in fact further limit the subject matter of the independent claims.

The Examiner has suggested relevant limitations in the independent and dependent claims that allegedly are redundant such that the dependent claims do not further limit the subject matter. However, the claim language (cited and otherwise) fails to indicate where in the independent claims, there exists any language regarding the generation of the information and when such information is generated. The dependent claims provide that such generation occurs at the time "when the element is transferred from the form to the HTML page associated with the HTML file." Accordingly, Appellants submit that the objection is improper and should be removed.

B. The Independent Claims are Patentable Over the Cited Art

The Final Office Action rejects claim 1 in an almost identical manner to that of prior rejections as follows:

In regard to independent claim 1, Foley teaches:

- project files within a portfolio file, said portfolio file containing references to members of a set of project files, said project file containing a URL of an HTML file including an applet tag (Foley

column 2 lines 55-63, column 8 lines 57-59, Figure 3 item 170A; compare with claim 1 "*reading information from a project file...*",

- Foley does not specifically teach a relationship between a form element and an HTML page and its associated HTML file. However, Francis teaches embedded form objects in an HTML page (said page processing a file name), whereby a relationship between form objects within said HTML page is generated with the help of a "Structured Language Element-to-Embeddable Object Class Association Table" (Francis column 4 lines 45-52, column 10, lines 53-64, column 14 lines 55-61; compare with claim 1 "*...the information comprising a relationship between an element that has been transferred from a form to an HTML page and the HTML file associated with the HTML page*", and "*from the form*"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Francis to Foley, because of Francis's taught advantage of defining relationships in order to provide editing of pages and forms within a single environment (as taught by Francis), to the single portfolio environment of Foley (See Francis column 4 lines 25-30).

- processing an applet referenced in each web document (Foley column 5, lines 32-49); compare with claim 1 "*processing the information to map the element to the HTML file*".

- a graphical user interface in the form of a Java Workshop, including presented icon specifications and a toolbar (Foley Figure 1, column 4, lines 28-43). Francis teaches a relationship between form objects within said HTML page as previously discussed, above. Foley does not specifically teach the visual display between mapped elements and an HTML file. However, Arora teaches the displayed mapping of elements to an HTML page (Arora Abstract, column 10 lines 60-65, column 14 lines 32-36, Figures 22, 42). Compare the above with claim 1 "*displaying the mapping on a graphical user interface that includes the relationship between the element, the form, and the HTML file*". It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Arora to Foley and Francis, because of the advantage of visibly showing mapped files, links, and objects of an HTML page in an organized fashion that Arora brings to Foley.

- It is noted that Applicant has amended claim 1 to recite "*...an element from a form, wherein the element is in an HTML page, comprising*". However, Francis still shows both a form, and elements of said form, within an HTML page (Francis column 5, lines 47-52, column 10 lines 17-23, 24-37, and 38-67). In addition, various form element tags are shown within the HTML code presented on column 10 lines 23-36, as well as element names of "FORM" and "INPUT", for the purpose of providing a form in a presented Web page. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Francis to Foley, providing Foley the benefit of creating project files for various Web pages (i.e., HTML pages combined (embedded) with input forms).

In response to Applicants' prior response that clearly set forth distinguishing factors, the Final Office Action responded as follows:

Applicant's arguments filed 7/11/2002 as paper No. 13, have been fully and carefully considered but they are not persuasive.

...

Applicant argues on page 12 to page 13 of the amendment that an HTML FORM tag is clearly distinguishable from an element that has been transferred from a form to an HTML page, a FORM tag element is not equivalent to a form from which elements are transferred into an HTML page. The examiner notes that a form as presently claimed can be interpreted as an input form which can be associated with HTML pages. Since a form is defined by HTML tag elements, the form and its tag data are transferred to said HTML page.

Applicant argues on page 13 of the amendment that Francis's Table is clearly distinguishable from information or a mapping that indicates the transfer from a form (and not an object or a class) to an HTML file associated with the HTML page, as claimed. The examiner notes that Francis teaches embedded form objects in an HTML page (said page possessing a file name), whereby a relationship between form objects within said HTML page is generated with the help of said Table.

Francis teaches a class identifier identifying an object selected according to a selected tag component (e.g., an HTML tag name) of the structure language element with reference to said Table (Francis column 4 lines 45-50). Francis teaches HTML tag names as form tag names (Francis column 10 lines 50-60, column 15 lines 40-50), and teaches "a relationship between an element that has been transferred from a form to an HTML page and the HTML file associated with the HTML page".

Applicant argues on page 13 of the amendment that Francis does not teach display in a graphical user interface of a mapping that include relationships. The examiner notes that Foley teaches project files. Francis teaches mapping between HTML form tag element and an HTML file, and Arora teaches the displayed mapping of elements to an HTML page (Arora column 14 lines 32-36, Figures 22, 42). In additional support of this rejection, Arora teaches a structure editor for Websites, including defining the layout of each page in a site (Arora Abstract). Arora also teaches a page editor including values of Properties window (properties for a displayed page) (Arora column 10 lines 61-67), and a list of draw objects (Arora column 11 lines 17-20).

Applicant argues on pages 13-14 the amendment that Foley and Francis do not teach a mapping of elements to an HTML file. The examiner notes that Foley teaches project files within a portfolio file, said portfolio file containing references to member of a set of project files, said project file containing a URL to an HTML file including an applet tag, and Francis teaches embedded form object in an HTML page (said page possessing a file name), whereby a relationship between form objects within said HTML page is generated with the help of a "Structured Language Element-to-Embeddable Object Class Association Table".

Applicant argues on pages 14-15 of the amendment that Arora does not teach display of a mapping of an HTML file. The examiner notes that Arora teaches the displayed mapping of elements to an HTML page.

In view of the above rejections, Applicants respectfully continue to traverse these rejections for one or more of the following reasons:

- (1) Neither Foley, Francis, nor Arora teach, disclose, or suggest the form as claimed;
- (2) Neither Foley, Francis, nor Arora teach, disclose, or suggest an element that has been transferred from a form to an HTML page;
- (2) Neither Foley, Francis, nor Arora teach, disclose, or suggest providing a mapping from an element transferred from a form to an HTML file associated with an HTML page;
- (3) Neither Foley, Francis, nor Arora teach, disclose, or suggest displaying a mapping from an element to an HTML file in a graphical user interface that indicates the relationship between the element, the form, and the HTML file;
- (4) Francis' form object is distinguishable from and not even remotely similar to the form from which elements are transferred from as claimed;

Each of these reasons will be discussed in further detail below.

As described above, the present claims provide for displaying (in a graphical user interface) a relationship (i.e., mapping) between an HTML file and an element from a form, wherein the element

was transferred to an HTML page. For example, a form can contain multiple elements. One element is transferred from the form to an HTML page (e.g., when creating the HTML page).

(1) The references fail to describe an element that has been transferred from a form to an HTML page as claimed.

As claimed, the graphical user interface indicates the relationship between the element, the form that the element was transferred from, and an HTML file that the element was transferred to. Such claim language illustrates that the user may easily view the relationships of objects involved in the creation of a web page in a graphical user interface. Such viewing capability may further allow a user to easily manipulate a page and various elements in a page using a tool for building an HTML page (associated with an HTML file). Further, as described in the specification, since the form contains one or more elements, and many instances of an element may be used in various HTML pages, only one copy of the form (that includes the element) needs to be retrieved locally (see page 15, lines 1-5). Accordingly, processing and transfer time are optimized.

Once the information is read from the project file that contains the relationship between the element and a particular HTML file, the information is processed to obtain a mapping from the element in the form to the HTML file. Once obtained, the mapping is displayed in a graphical user interface. The graphical user interface allows the user to identify the element in the HTML file, the form where the element was transferred from, and the HTML file where the element was transferred to.

The Office Action admits that Foley fails to teach the relationship between a form element and an HTML page and its associated HTML file and utilizes Francis to teach this claim element. The Office Action cites col. 10, lines 53-64 to teach the form as claimed. Col. 10, lines 53-64 merely refers to an HTML FORM tag element. Essentially, Francis's HTML FORM tag element is utilized to teach the element transferred from a form as claimed. However, as described herein, an HTML FORM tag element is clearly distinguishable from an element that has been transferred from a form to an HTML page. While the element transferred could be a FORM tag element, a FORM tag element is not equivalent to a form from which elements are transferred into an HTML page.

The examiner submits that the claimed form can be an input form that can be associated with HTML pages. The examiner further submits that since a form is defined by HTML tag elements, the form and its tag data are transferred to said HTML page.

Appellants respectfully disagree. As claimed, the element is transferred "from the form to the HTML page". Transferring a form and its tag data to an HTML page is not equivalent to transferring an element from a form to an HTML page. Specifically, it is well known in the art (and consistently described in Francis) that the FORM tag element is used to create a form in an HTML page that the user can fill in. Such a meaning is clearly distinguishable from the form in the present claims wherein an element is transferred from a form to an HTML page. This particular meaning of Francis' FORM is described at col. 10, line 18 through col. 11, line 7. This portion of Francis illustrates that a particular HTML element has the name FORM and an HTML document may have several such FORM elements. The various elements are used to create a form that the user can interact with. This meaning is further admitted by the Examiner on page 4 of the Final Office Action where the Examiner states "In addition, various form element tags are shown within the HTML code presented on column 10 lines 23-36, as well as element names of 'FORM' and 'INPUT', for the purpose of providing a form in a presented Web page." The present claims have nothing to do with providing a form in a presented Web page. Instead, the claims are directed towards creating an HTML page using elements from a form and later displaying a relationship between the elements and the form from which they were transferred.

Francis completely fails to describe transferring any element from a form (and not a FORM element) to an HTML page. Further, while Francis' FORM may contain an element in an HTML page, Francis' FORM elements are not transferred from a form (containing numerous elements) to the HTML page (as claimed). In this regard, it is logically impossible in Francis to transfer an element from the FORM to an HTML page as claimed.

(2) The references fail to describe providing or displaying a mapping from an element to an HTML file in a graphical user interface that indicates the relationship between the element, the form, and the HTML file:

The Final Office Action uses Francis' "Structured Language Element-to-Embeddable Object Lass Association Table" (hereinafter referred to as "Table") to provide for the relationship between the element (that has been transferred from the form to the HTML page) and the HTML file associated with the HTML page. Col. 16, lines 43 - col. 17, line 8 describes such a Table. As described in Francis, the Table merely comprises an HTML tag component 226 with an object class identifier (CLSID) 234 (see col. 16, lines 48-50). Accordingly, the Table is merely a list of tag components and an identifier for an object used to create the component. As further illustrated in Francis, when an HTML tag is identified/selected while trying to transform an HTML document into RTF format (see col. 15), the Table is searched for the selected tag component (see col. 16, lines 64-66). Thus, the Table in Francis is a relationship between a tag and an object used to create the tag and not an element and a form and an HTML file. Further, there is no indication in Francis that the class in the Table is from a form.

Such a Table is clearly distinguishable from information or a mapping that indicates the transfer from a form (and not an object or a class) to an HTML file associated with the HTML page (as claimed). Further, contrary to that suggested in the Office Action, a listing of objects and tags is NOT equivalent to a relationship between an element that has been transferred from a form to an HTML page and the HTML file associated with the HTML page. Thus, while Francis describes tags and class ids, there is no fully displayed mapping that indicates the form where an object/element is obtained from or the HTML file that the object/element is placed in.

Additionally, the Table of Francis is used to convert HTML to RTF (see col. 14, line 56 - col. 15 line 2). However, in the present invention, the end result is the display in a graphical user interface of a mapping that indicates the relationships. Such a display may then be manipulated and used as the user desires. Such a method, use, and result are clearly distinguishable from that of Francis.

In response to the above (as presented in an earlier submission by Appellants), the Examiner asserts that Francis teaches embedded from objects in an HTML page and relationship between

such embedded form objects is generated with the help of a table. However, Appellants submit that a relationship between objects within an HTML page is clearly distinguishable from a relationship between an element in an HTML page and the form that originally contained the element (as claimed). Further, since Francis fails to describe an element that has been transferred from a form to an HTML page, Francis cannot possibly describe or display a mapping that illustrates such a relationship.

(3) The references fail to describe processing information to map the element from the form to the HTML file.

The Office Action submits that Foley provides for the second element of the independent claims (i.e., "processing the information to map the element from the form to the HTML file"). However, while Foley may provide for processing an applet referenced in each web document (as indicated in the Office Action), such processing does not result in mapping the element to the HTML file. In Foley, the processing merely provides that the JWS browser pulls in and begins executing any referenced applets found in a Web document (see col. 5, lines 9-31). However, the present claims provide that the processing results in mapping an element that has been transferred from a form to an HTML page and not beginning the execution of the applets. Execution of an applet is clearly different from mapping. Accordingly, Foley fails to teach, disclose, or suggest the invention as claimed.

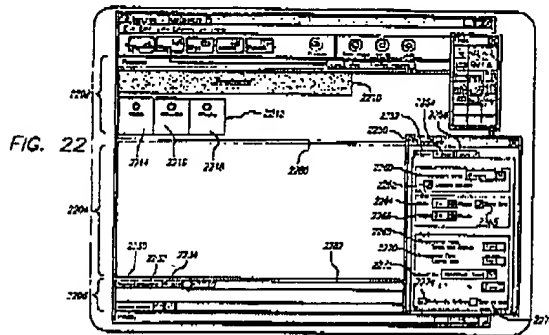
Additionally, even if Foley were combined with Francis, the result would be to process the applets in a web document (from Foley) and add them to a table of applets and their corresponding tags (as in Francis). Such a teaching is clearly different from reading information and processing to obtain a mapping from the element to an HTML file.

In response to the above arguments (submitted in a prior reply by Appellants), the Final Office Action alleges that Foley teaches project files within a portfolio file that contains references to members of a set of project files and further containing a URL of an HTML file including an applet tag. The final Office Action then reiterates that Francis teaches embedded form objects in an HTML page. Such assertions do not contradict Appellants' traversal. Nowhere in Foley, Francis, or Arora is there any teaching of a mapping whatsoever. In this regard, Foley's file that contains

references to a set of project files where a project file contains a URL of an HTML file including an applet tag in combination with Francis' HTML file that contains embedded FORM objects is not even remotely similar to mapping an element from a form to an HTML file as claimed.

(4) The cited references fail to describe displaying a mapping as claimed.

The Office Action rejects the last claim element under Arora FIGS. 22 and 42 and col. 14, lines 32-36 and indicates that Foley fails to teach this element. However, as indicated in the prior Office Action responses, FIGS. 22 and 42 and col. 14, lines 32-36 of Arora provide for an Assets Display that shows the files, links, and objects in a data processing system.



Additionally, merely displaying the values of a properties window (Arora column 10, lines 61-67) (i.e., a grid of pixels, the font used and the colors, etc.) and a list of draw objects (Arora col. 11 lines 7-20) (i.e., a list of objects on a page) is not even remotely similar to displaying a mapping that indicates the relationship between an element, a form (that the element was transferred from), and an HTML file (as claimed).

Accordingly, Arora fails to teach, disclose, or suggest the invention as claimed.

In response to the above arguments (submitted by Appellants in a prior response), the Final Office Action merely states: "The examiner notes that Arora teaches the displayed mapping of elements to an HTML page". Appellants respectfully traverse such a "notation" as it fails to recognize the differences as described herein.

None of the cited references provides for all of the capabilities and details as claimed. Further, as described above, when combined, the references actually teach away from Applicants' invention. Additionally, the various elements of Applicants' claimed invention together provide operational advantages over the systems disclosed in Foley, Arora, Francis, and Lisle. In addition, Applicants' invention solves problems not recognized by Foley, Arora, Francis, and Lisle.

Thus, Applicants submit that independent claims 1, 12, 23, and 34 are allowable over Foley, Francis, and Arora.

C. Dependent claims 2, 13, and 24 are Patentable over the Prior Art

Dependent claims 2, 13, and 24 provide a timing element that specifies when the information in the independent claims is generated. Specifically, the information is generated when the element is transferred from the HTML page associated with the HTML file. None of the cited references teach, disclose, or suggest such a timing element as claimed. Thus Appellants submit that dependent claims 2, 13, and 24 are allowable over the cited references.

D. Dependent claims 3, 14, and 25 are Not Separately Argued

E. Dependent claims 4, 15, and 26 are Not Separately Argued

F. Dependent claims 5, 16, and 27 are Patentable over the Prior Art

Dependent claims 5, 16, and 27 depend on claims 4, 15, and 26 and further add the limitation that the information also comprises a form name. In rejecting these claims, the Final Office Action submits a form name would be obvious in view of Francis' teaching that an HTML element can have a name "FORM". Appellants respectfully traverse such a suggestion. Having an element named FORM is not even remotely similar to a form that has a name. Such an analogy is equivalent to stating that having a computer named Apple teaches an apple that has a name (e.g., Granny Smith, Golden Delicious, etc.).

In addition, this claim further establishes that the claimed form is distinguishable from the FORM element described in Francis. Thus Appellants submit that dependent claims 5, 16, and 27 are allowable over the cited references.

G. Dependent claims 6, 17, and 28 are Not Separately Argued

H. Dependent claims 7, 18, and 29 are Patentable over the Prior Art

Dependent claims 7, 18, and 29 further elaborate on the table of claims 6, 17, and 28. Specifically, these claims indicate that a table has cells that are defined by row and column and the mapping is entered into the cell. The Final Office Action relies on Arora Figure 39 to teach this claim element. The description of Arora's Figure 39 (col. 13, line 57 – col. 14, line 5) provides:

FIG. 39 is an example of a matrix 3802 generated in accordance with the edges and draw objects. Matrix 3902 has a number of rows equal to the number of row edges plus one. The matrix has a number of columns equal to the number of column edges plus one. Thus, in the example, the matrix has seven columns and eight rows. Each element 3904 of the matrix has a pointer field and an occupied flag. The element of row 7, column 2 of the matrix points to object number 1 and is marked as occupied. The next four elements in row 2 are also marked as occupied. The element of row 3, column 3 points to object number 2 and is marked as occupied. The element of row 2, column 5 points to object number 3 and is marked as occupied. The next four elements in column 5 are also marked as occupied. The element of row 5, column 3 points to object number 4 and is marked as occupied.

Thus, Figure 38 merely describes cells that has a pointer field and an occupied flag. In this regard, there is no mapping entered into a particular cell. Further, Arora completely fails to describe the use of such a mapping or table.

Thus, Appellants submit that dependent claims 7, 18, and 29 are allowable over the cited references.

I. Dependent claims 8, 19, and 30 are Not Separately Argued

J. Dependent claims 9, 20, and 31 are Not Separately Argued

K. Dependent claims 10, 21, and 32 are Patentable over the Prior Art

Dependent claims 10, 21, and 32 further elaborates the flagging of claims 9, 20, and 31. Specifically, a project file is read and the HTML file name is extracted from the project file. Thereafter, the HTML file name is used to search for the HTML file. If the HTML file is not found, the mapping is flagged as invalid.

In rejecting this claim, the Office Action provides: "claim 10 incorporates substantially similar subject matter as claimed in claims 1 and 9, and is rejected along the same rationale."

However, claims 1 and 9 do not provide for reading a project file; claims 1 and 9 do not provide for extracting the HTML filename from the project file; Claims 1 and 9 do not provide for searching for the HTML file using the extracted HTML filename; and claims 1 and 9 do not provide for flagging the mapping as invalid when the extracted HTML file is not found. In fact, every step of these independent claims are substantially different from the claims on which they depend. Lisle fails to teach each aspect of these claims. In addition, the remaining references also fail to teach the claimed elements.

Thus, Appellants submit that dependent claims 10, 21, and 32 are allowable over the cited references.

L. Dependent claims 11, 22, and 33 are Not Separately Argued

IX. Conclusion

In light of the above arguments, Appellant respectfully submits that the cited references do not anticipate nor render obvious the claimed invention. More specifically, Appellant's claims recite novel physical features which patentably distinguish over any and all references under 35 U.S.C. §§ 102 and 103. As a result, a decision by the Board of Patent Appeals and Interferences reversing the Examiner and directing allowance of the pending claims in the subject application is respectfully solicited.

Respectfully submitted,

GATES & COOPER LLP

Attorneys for Applicant(s)

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: February 28, 2003

By: GHG/JSF

Name: George H. Gates

Reg. No.: 33,500

GHG/JSF

G&C 30571.179-US-01

APPENDIX

1. A method of displaying a relationship between an HTML file and an element from a form, wherein the element is in an HTML page, comprising:
reading information from a project file, the information comprising a relationship between the element that has been transferred from the form to the HTML page and the HTML file associated with the HTML page;
processing the information to map the element from the form to the HTML file; and
displaying the mapping on a graphical user interface that indicates the relationship between the element, the form, and the HTML file.
2. The method of claim 1, wherein the information is generated when the element is transferred from the form to the HTML page associated with the HTML file.
3. The method of claim 1, wherein the element is selected from a group comprising:
a visual control; and
a non-visual control, selected from a group comprising a button, a picklist, and a data entry box.
4. The method of claim 1, wherein the information comprises an element name and an HTML file name.
5. The method of claim 4, wherein the information further comprises a form name.
6. The method of claim 1, wherein the displaying the mapping comprises presenting an element name and an HTML file name in a row of a table.
7. The method of claim 6, wherein the table comprises cells defined by the row and the column of the cell, and the mapping is entered into a cell of the table.

8. The method of claim 1, wherein the displaying the mapping comprises presenting the element name and the HTML file name in a column of a table.

9. The method of claim 1, further comprising flagging an invalid mapping between the element and the HTML file.

10. The method of claim 9, wherein the flagging an invalid mapping between the element and the HTML file comprises:

- reading the project file;
- extracting the HTML filename from the project file;
- searching for the HTML file using the extracted HTML filename; and
- flagging the mapping as invalid when the extracted HTML file is not found.

11. The method of claim 1, further comprising:

- accepting a modified mapping; and
- storing the modified mapping in the project file.

12. A apparatus for displaying a relationship between an HTML file and an element from a form, wherein the element is in an HTML page, comprising:

- means for reading information from a project file, the information comprising a relationship between the element that has been transferred from the form to the HTML page and the HTML file associated with the HTML page;

- means for processing the information to map the element from the form to the HTML file;
- and

- a display for presenting the mapping to a user on a graphical user interface that indicates the relationship between the element, the form, and the HTML file.

13. The apparatus of claim 12, wherein the information is generated when the element is transferred from the form to the HTML page associated with the HTML file.

14. The apparatus of claim 12, wherein the element is selected from a group comprising:
a visual control;
a non-visual control, selected from a group comprising a button, a picklist, and a data entry box.
15. The apparatus of claim 12, wherein the information comprises an element name and an HTML file name.
16. The apparatus of claim 15, wherein the information further comprises a form name.
17. The apparatus of claim 12, wherein the means for displaying the mapping comprises means for presenting the element name and the HTML file name in a row of a table.
18. The apparatus of claim 17, wherein the table comprises cells defined by the row and the column of the cell, and the mapping is entered into a cell of the table.
19. The apparatus of claim 12, wherein the means for displaying the mapping comprises means for presenting the element name and the HTML file name in a column of a table.
20. The apparatus of claim 12, further comprising means for flagging an invalid mapping between the element and the HTML file.
21. The apparatus of claim 20, wherein the means for flagging an invalid mapping between the element and the HTML file comprises:
means for reading the project file;
means for extracting the HTML filename from the project file;
means for searching for the HTML file using the extracted HTML filename; and
means for flagging the mapping as invalid when the extracted HTML file is not found.
22. The apparatus of claim 12, further comprising:
means for accepting a modified mapping; and

means for storing the modified mapping in the project file.

23. An article of manufacture, embodying logic to perform a method of displaying a relationship between an HTML file and an element that has been transferred from a form to an HTML page, the method comprising:

reading information from a project file, the information comprising a relationship between an element that has been transferred from a form to an HTML page and the HTML file associated with the HTML page;

processing the information to map the element from the form to the HTML file; and

displaying the mapping on a graphical user interface that indicates the relationship between the element, the form, and the HTML file.

24. The article of manufacture of claim 23, wherein the information is generated when the element is transferred from the form to the HTML page associated with the HTML file.

25. The article of manufacture of claim 23, wherein the element is selected from a group comprising:

a visual control; and

a non-visual control, selected from a group comprising a button, a picklist, and a data entry box.

26. The article of manufacture of claim 23, wherein the information comprises an element name and an HTML file name.

27. The article of manufacture of claim 26, wherein the information further comprises a form name.

28. The article of manufacture of claim 23, wherein the displaying the mapping comprises presenting the element name and the HTML file name in a row of a table.

29. The article of manufacture of claim 28, wherein the table comprises cells defined by the row and the column of the cell, and the mapping is entered into a cell of the table.

30. The article of manufacture of claim 23, wherein the displaying the mapping comprises presenting the element name and the HTML file name in a column of a table.

31. The article of manufacture of claim 23, wherein the method further comprises flagging an invalid mapping between the element and the HTML file.

32. The article of manufacture of claim 31, wherein the flagging an invalid mapping between the element and the HTML file comprises:

reading the project file;

extracting the HTML filename from the project file;

searching for the HTML file using the extracted HTML filename; and

flagging the mapping as invalid when the extracted HTML file is not found.

33. The article of manufacture of claim 23, wherein the method further comprises:
accepting a modified mapping; and
storing the modified mapping in the project file.

34. A computer readable data structure for representing a software project in a single file, the software project comprising a project application defined by executable programming logic, and a project environment for developing the application, the data structure comprising:

a first section comprising the executable programming logic needed to load and execute the project application in the computer; and

a second section for storing data required to restore the project environment, and for storing information comprising a relationship between elements that have been transferred from a form to an HTML page and HTML files associated with the HTML page in the project;

wherein the relationship between elements, the form, and the HTML file is displayed in a graphical user interface.